

Progettazione di Applicazioni Web T
Prova d'Esame – 30 giugno 2022

Tempo a disposizione: 2 ore e 30 minuti

La soluzione comprende la consegna elettronica dei seguenti file:

Cineteca.zip	file zip contenente il sorgente java/class, file XML/XSD e txt per il punto 1
Pallacanestro.zip	file zip contenente il sorgente java/class e file txt per il punto 2
WebRatio.zip	file zip contenente il testo di risposta (txt/doc/pdf) al punto 3

Ogni file .zip consegnato DEVE CONTENERE TUTTI e SOLI i file creati/modificati e/o ritenuti importanti in generale ai fini della valutazione (ad esempio, codice java e relativi .class, descrittori XML, file txt/doc/pdf, ecc.) e NON dell'intero progetto.

N.B. La prova si intende superata se il punteggio finale non è inferiore a 18 punti (su un totale di 33). In particolare, è necessario (ma non sufficiente) ottenere la sufficienza in tutti gli esercizi.

ESERCIZIO 1 - 14 punti (sufficienza: 7 punti)

Si progetti una **grammatica XML Schema**, assieme a un suo **documento XML** di esempio, in grado di modellare le informazioni relative **all'archivio film della Cineteca di Bologna**, nel rispetto delle seguenti specifiche:

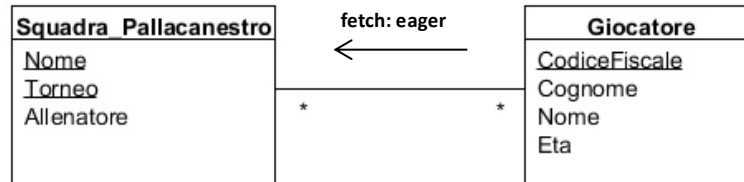
- L'archivio film si divide in due tipologie di contenuti: l'archivio audiovisivo costituito da 31.000 filmati e l'archivio pellicola (contenente 60.000 pellicole).
- A fini di ricerca nel database, ogni audiovisivo è caratterizzato dalla coppia [*Cognome, Nome*] del regista e da un titolo [*Titolo italiano*].
- Ogni audiovisivo ha inoltre associato informazioni quali la nazione (es., "IT"), la lingua (es., "ITA"), il tipo di colore, ovvero {*Colore, BN*}, la durata in minuti, la tipologia (es., "Documentario"), il formato specifico (nel dettaglio, {*Blu-ray, DVD, Divx, VHS*}), l'anno di produzione, la disponibilità per la consultazione (*si/no*) e la collocazione fisica all'interno della cineteca. Nel dettaglio, la collocazione è rappresentata da un codice complesso formato da 5 componenti separate dal carattere "."; nel dettaglio, le prime tre parti del codice sono formate da due cifre intere, la quarta è formata da 3 cifre intere, mentre l'ultima è formata da una stringa di 3 caratteri (a titolo di esempio, "05.02.20.053.BIB").
- Per quanto riguarda le pellicole, le stesse possono essere richieste solo in forma "manuale", ovvero mediante lettera scritta, indicando "regia" e "titolo" del film di interesse; le pellicole possono essere ammesse al prestito se, previa revisione, e formulazione di un esito (*si/no*), risultino essere in buono stato di conservazione. Le pellicole possono essere di due tipologie (pellicole in 35 e 16 mm).

Si realizzi quindi l'**applicazione Java "Cineteca"** che, facendo uso del **parser DOM** e del **documento XML di esempio** realizzato al punto precedente, esponga i metodi `getAudiovisiviBN()` e `getPellicole16()`, unitamente a un suo `main` di prova, in grado di restituire: (i) quanti audiovisivi in bianco e nero sono presenti nella cineteca e (ii) quante pellicole in 16 mm sono in buono stato di conservazione presso la cineteca. Stampare il risultato prodotto dai metodi sul file **Cineteca.txt**.

Progettazione di Applicazioni Web T
Prova d'Esame – 30 giugno 2022

ESERCIZIO 2 - 14 punti (sufficienza: 7 punti)

Partendo dalla realtà illustrata nel **diagramma UML** di seguito riportato, si fornisca una soluzione alla gestione della persistenza basata su **metodologia DAO** in grado di “mappare” efficientemente, e con uso di ID surrogati, il modello di dominio rappresentato dai **JavaBean Squadra_Pallacanestro** e **Giocatore** del **diagramma UML** con le corrispondenti **tabelle relazionali derivate dalla progettazione logica** del diagramma stesso. Nel dettaglio, la gestione degli ID surrogati è demandata a un'apposita classe `IdBroker()` la cui logica si deve basare sull'uso di campi “auto increment” (ovvero, `identity column` in DB2).



N.B. Relativamente allo schema UML: (Nome, Torneo) è **chiave** per la relazione `Squadra_Pallacanestro`; CodiceFiscale è **chiave** per la relazione `Giocatore`.

Nel dettaglio, dopo aver creato da applicazione Java gli **schemi delle tabelle** all'interno del proprio schema nel database `TW_STUD` di **DB2** (esplicitando tutti i vincoli opportuni), implementato i **JavaBean** e **realizzato le classi** relative al **Pattern DAO** per l'**accesso CRUD** alle tabelle, si richiede l'**implementazione di un metodo** in grado di restituire: *“Per ogni giocatore, i nomi degli allenatori delle squadre in cui ha giocato”*.

Si richiede quindi di realizzare una **classe di prova** in grado di:

- inserire due o più tuple nelle tabelle di interesse al fine di produrre un risultato consistente rispetto a quanto richiesto dal metodo al punto precedente;
- fare uso corretto del metodo realizzato al fine di produrre la stampa del risultato, opportunamente formattata, sul file **Pallacanestro.txt**;

Il candidato è invitato a motivare opportunamente le proprie scelte in fase di modellazione dei JavaBean e della creazione delle tabelle derivate dal passaggio di progettazione logica, mediante commento da riportare sullo stesso file Pallacanestro.txt.

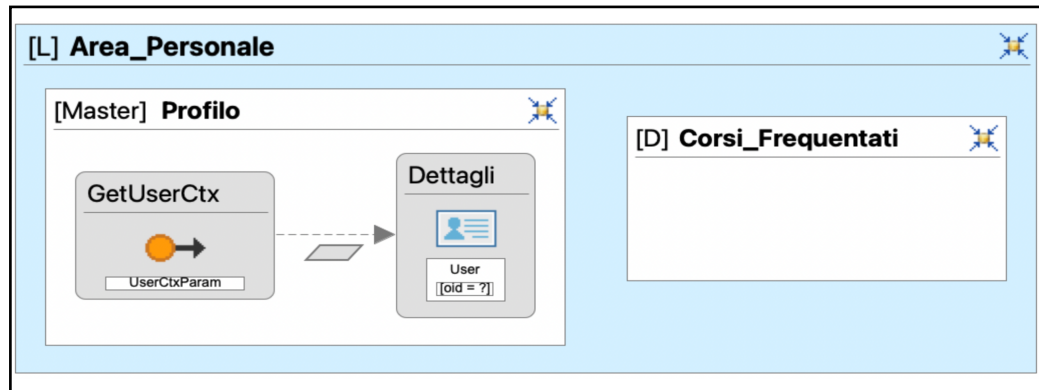
N.B. L'implementazione **deve limitarsi** al solo **DBMS DB2**. La soluzione **deve sfruttare il mapping N-M specificato nello schema UML e propendere per il caricamento dei dati indicato nello stesso**. Ogni ulteriore scelta da parte dello studente deve essere opportunamente giustificata con commenti nel codice.

Progettazione di Applicazioni Web T
Prova d'Esame – 30 giugno 2022

ESERCIZIO 3 - 5 punti (sufficienza: 3 punti)

Considerando l'ipertesto progettato nello **schema IFML di WebRatio** di seguito riportato,

SV



il candidato descriva dettagliatamente le relazioni esistenti tra le componenti area, pagina, link e condizioni espresse sulle componenti della *site view* **SV**.